Supporting C2 with a Service Oriented Framework for Opportunistic Sensors and Sensor Networks

Marianela Garcia Lozano,
Pontus Hörling,
Farshad Moradi,
Edward Tjörnhammar
Outline

- C2 challenges and vision
- Related research
- End user Workshop and results
- TOppS Objectives and Architecture
- Framework design
- Summary and conclusions
- Questions, discussion
C2 Challenges

- Survivability
- Rapid development
- Evaluation
- Interoperability
- Automation
- Sensor
  - situation-dependent behaviour
    - calibration, choice of sensing modes, data fusion, inter sensor queuing and communication, establish connections C2 systems, remote sensor management
Workshop Scenario
Workshop Results

- Requests for a more modular system (both in software and hardware).
- Possibility to incorporate legacy sensors as well as new ones.
- Ease of use.
- Plug and Play (automatic sensor configuration, identification and calibration).
- Web of trust amongst sensor producers.
- Light weight, easy to deploy sensors.
- Sensors with low power consumption.
- Better data control i.e. meta data needed about provided data.
Related Research

- Opportunistic Sensors
- NECC
- OGC, SWE
- SOA
- Pervasive computing
- SODA
Objectives

- To allow opportunistic use of sensors
- To view sensors as web services
- To design and implement a service based framework for Opportunistic Sensors and sensor networks
- To test and evaluate methods and techniques for:
  - querying sensor services
  - composing sensor services
  - achieving plug and play of sensors
  - achieving (location, type) transparency in sensor use
TOppS Vision

Every sensor:
- Plug & Play
- Where and When am I?
- What do I see?
- How am I?
- Availability
- History

Network of networks

Services
- Do we have coverage?
- Is the sensor network operational?
- Surveillance and Collaboration
- Direction of sensors

Locally
Globally
Modules: Device Launcher

- Resides close to the Operating System and the specific hardware.

- Provides for simple adaptation of a sensor to the framework to facilitate basic communication, i.e. bootstraps the sensor into the framework.

- Is specialized and aligned to the communication protocol of a specific sensor type i.e. Bluetooth, USB, Ethernet, etc.
Modules: Presenter

- Presents the sensor as a service and describes its communication semantics, i.e. REST.
- Provide an access point for data and instruction streams.
- Publishes sensor events and data with RSS.
- Sends produced data for logging.
Modules: Logger

- Provides a temporary storage of sensor data
- Is implemented as a cyclic buffer in local mode with persistence constraints.
- Loggers can be distributed or centralized.
Module: Broker (Publisher/Service Discovery)

- Broker provides a matching between user needs and available services.
- It provides a common interface towards other components.
- It consists of two modules:
  - The publisher registers new services with the Registry.
  - The Service Discovery makes a lookup in the Registry for required services.
Modules: Registry & Composer

- **The Registry:**
  - keeps track of published services
  - performs garbage collection
  - might also be a distribution of registries

- **The Composer:**
  - translates the user requests into service types and service properties
  - queries the Service Discovery module
  - composes a new service if necessary
Composition process

- Receive and analyze requests from a user (via a GUI)
- Discover suitable services
- Possibly combine discovered Services into a new Service and store it in the registry for later use
Composition concept examples

- **Alarm service e.g.** provide me with a video stream when someone enters room 111
  - Two heterogeneous sensors: motion detection and camera
  - One triggers the other

- **Speed service e.g.** detect vehicle presence at two different points and calculate speed
  - Two homogeneous sensor services and one calculation service

- **Coordinate transformation e.g.** give someone’s location in long – lat from sensor data in x, y, z
  - One location detection sensor and one transformation service
Modules: Security

- Authenticares and Authorize users and services (via Spring Security/LDAP)

- Three main points of interaction:
  - between the GUI and the composer
  - the service presenter and the Broker
    - Publishing services
    - Discovering services

- Sensor data is owned by producer
Map View (Sensor Overview)
Test and evaluation

- User workshop 10\textsuperscript{th} of June.
- The purpose is/was to test:
  - The framework
  - Discovery of services
  - Composition of services
  - Sensor information requests
  - Plug and play functionality of sensors
Second Workshop
Summary

Presented:

- C2 challenges.
- Our vision.
- First workshop and results.
- System outline for an opportunistic sensor-service framework and how it could be used to support C2 systems with opportunistic sensor data.
- Results and current status.
Questions?

→ Thank you!